

REMARKS

Claims 1-11, 14-20, 23-29, 32 and 35-36 are now pending in this application. Claim 1-6 and 8-11 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over reference V on PTO-892 mailed August 31, 2007 (“CTRC”) in view of PTO-892 mailed August 22, 2006, August 31, 2007 and November 14, 2007 (“Taylor”), PTO-892 mailed November 14, 2007 (“Coulouris”) and United States patent Number 6,347,341 B1 (“Marcos”). Claims 23-29, 32, 35 and 36 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over CTRC in view of Taylor, Coulouris, Marcos and United States Patent number 6,442,663 B1 (“Sun”). Claim 7 stands rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over CTRC in view of Taylor, Coulouris and Marcos and in further view of United States Patent number 6,230,117 B1 (“Lymer”). Claims 14-20 stand rejected under 35 U.S.C. § 103(a) as allegedly being upatentable over Marcos in view of CTRC, Taylor and Coulouris.

Applicants respectfully traverse. Claims 1, 14, 23 and 32 have been amended.

Rejection of Claims 1-11 Under 35 U.S.C. § 103(a)

Claims 1-6 and 8-11 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over CTRC in view of Taylor, Coulouris and Marcos. Claim 7 stands rejected under 35 U.S.C. § 103(a) as allegedly being upatentable over CTRC in view of Taylor, Coulouris and Marcos and in further view of Lymer.

Claim 1 as amended recites a method of enabling a first system to use a second system comprising *receiving at least one data declaration pertaining to the first system, generating interoperability information for the first and second systems comprising, an interface definition language (“IDL”) description of a portion of code on the first system for interacting with the second system, at least one class definition pertaining to the at least one data declaration of the first system and storing the interoperability information.*

Claim 1 further recites receiving at the first system, a first request directed to the second system . . . performing a first conversion of said first request to produce a second request *using the interoperability information*, said second request being in a form adapted for said second system but not for said first system, said first conversion performed using a

host initiated processing (HIP) system that includes one or more listeners for receiving the first request and one or more HIP proxies *for instantiating at least one flow control each of the at least one flow control associated with a respective requesting port* for the first conversion . . . and performing a second conversion of said first reply *using the interoperability information* to produce a second reply that comprises header information that is usable with an Internet protocol . . .

Support for this amendment may be found, for example, on pages 14-21 of the specification. Briefly summarizing this discussion, FIG. 6 shows an exemplary system 600 for providing interoperability between heterogeneous computer systems, such as computer 201 and computer 202. The system 600 may comprise among other components a HIP runtime service 602, a listener, a request queue, a HIP proxy, a flow control, an invoker, conversion components, a send/receive object and a request context.

A flow control 610 is responsible for executing the plan that converts requests from computer 201 into a form suitable for use by applications on computer 202. This plan is defined at development time. Different flow controls are designed to handle different types of requests, which can be identified by the port on which they come in. For example, it may be defined that any request that comes in on port 1 is handled by one flow control, and request that comes in on port 2 is handled by a different flow control, etc. As one example, a flow control may be an "end point resolution" flow control that simply causes a particular method on a particular object (in computer 202) to be executed in response to a request received from computer 201

After code has been written for mainframe 704, at some point in the future developer 708 undertakes the task of enabling the mainframe code to call a component in a Windows-based system. Developer 708 uses component builder 710 in order to create the information that will aid in the conversion of data and programming model from the mainframe environment to the Windows environment. In particular, the component builder 710 generates an IDL (Interface Definition Language) 712 description of the portion of the mainframe code that will need to interact with Windows, and also generates Visual Basic class definitions 714 corresponding to the mainframe code's data declarations 706. It should be noted that some data types in Visual Basic may correspond very closely to data types in typical mainframe languages (such as COBOL or RPG), and others may not. For example,

Visual Basic and COBOL both have integers, but COBOL has dependent-sized arrays and Visual Basic does not. After the information that will support interoperability has been created, it is stored in a server component 716 for use by HIP service 602.

FIGS. 8-9 show the end-to-end experience of an HIP system from both the development and user perspective. Mainframe 201 is a non-WINDOWS system (e.g., a legacy system), and it is desired to allow applications on mainframe 201 to call software objects in a WINDOWS-based environment. A developer 802 uses a component builder tool 810 to prepare components that will allow an application on mainframe 201 to call software objects in a WINDOWS-based system. Developer 802 uses component builder tool 810 to analyze data declarations 706 that are associated with the legacy application. Component builder 810 allows developer 802 to use the above-mentioned information to create conversion information that will allow the HIP system to convert a call from mainframe 201 into a form that is usable by a WINDOWS-based software object. The conversion information produced by component builder 810 is stored in a .TIM file 818. The substance of this conversion information includes class definitions 812 and data declarations 816. Class definitions 812 may be later used in a developer's studio 814.

After the conversion information has been stored in a TIM file 818, an administrator 824 may prepare the HIP system to work with the application that developer 802 analyzed with component builder 810. Administrator 824 uses an administrative console 822 to install the conversion information contained in the .TIM file 818 into database 634.

Neither CTRC, Taylor, Coulouris nor Marcos taken alone or in combination teach or suggest receiving at least one data declaration pertaining to the first system, generating interoperability information for the first and second systems comprising, an IDL description of a portion of code on the first system for interacting with the second system, at least one class definition pertaining to the at least one data declaration of the first system and storing the interoperability information as recited in amended claim 1.

Furthermore, none of the cited references alone or in combination teach or suggest performing a first conversion of said first request to produce a second request using the interoperability information as required by amended claim 1.

CTRC, Taylor, Coulouris and Marcos further fail to disclose at least one flow control associated with a respective requesting port for the first conversion and performing a second

conversion of said first reply using the interoperability information to produce a second reply that comprises header information that is usable with an Internet protocol as recited in claim 1.

As the cited references fail to teach or suggest the amended claim elements, claim 1 should be allowed. Claims 2-11 depend from and therefore include all the limitations of claim 1. Therefore, for at least the reasons stated with respect to claim 1, claims 2-11 should also be allowed.

Rejection of Claims 14-20 Under 35 U.S.C. § 103(a)

Claims 14-20 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Marcos in view of CTRC, Taylor and Coulouris.

Claim 14 as amended includes limitations similar to amended claim 1 including *receiving at least one data declaration pertaining to the first system, generating interoperability information for the first and second systems comprising an interface definition language (“IDL”) description of a portion of code on the first system for interacting with the second system at least one class definition pertaining to the at least one data declaration of the first system and storing the interoperability information and generating conversion information using the interoperability information.*

Thus, for at least the reasons stated with respect to claim 1, claim 14 should be allowed. Claims 15-20 depend from and therefore include all the limitations of claim 14. Therefore, for at least the reasons stated with respect to claim 14, claims 15-20 should also be allowed.

Rejection of Claims 23-29 Under 35 U.S.C. § 103(a)

Claims 23-29 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over CTRC in view of Taylor, Coulouris, Marcos and Sun.

Claim 23 as amended includes limitations similar to claim 1 and 14 in system form including hardware comprising at least one processor, *a component builder module that operates on said hardware, said component builder receiving at least one data declaration pertaining to the first system, generating interoperability information for the first and second systems comprising an interface definition language (“IDL”) description of a portion of code*

on the first system for interacting with the second system, at least one class definition pertaining to the at least one data declaration of the first system and storing the interoperability information.

Claim 23 as amended further recites *a service object that executes on said hardware and that receives a first request from the first software object, converts the first request into a second request which is in a form usable by the second software object, and presents the second request to the second software object using the interoperability information.*

Thus, for at least the reasons stated with respect to claim 1, and claim 14, claim 23 should be allowed. Claims 24-29 depend from and therefore include all the limitations of claim 23. Therefore, for at least the reasons stated with respect to claim 23, claims 24-29 should also be allowed.

Rejection of Claims 32 and 35-36 Under 35 U.S.C. § 103(a)

Claims 32 and 35-36 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over CTRC in view of Taylor, Coulouris, Marcos and Sun. Claim 32 as amended recites a computer readable storage medium comprising instructions and includes limitations similar to claim 1, 14 and 23 *including receiving at least one data declaration pertaining to the first system, generating interoperability information for the first and second systems comprising, an interface definition language (“IDL”) description of a portion of code on the first system for interacting with the second system, at least one class definition pertaining to the at least one data declaration of the first system and storing the interoperability information.*

Thus, for at least the reasons stated with respect to claim 1, claim 14, and claim 23, claim 32 should be allowed. Claims 35-36 depend from and therefore include all the limitations of claim 32. Therefore, for at least the reasons stated with respect to claim 32, claims 35-36 should also be allowed.

DOCKET NO.: 303999.1 / MSFT-1961
Application No.: 10/610,689
Office Action Dated: September 30, 2008

PATENT

CONCLUSION

In view of the above amendments and remarks, applicant respectfully submits that the present invention is in condition for allowance. Reconsideration of the application is respectfully requested.

Date: December 30, 2008

/Kenneth R. Eiferman/

Kenneth R. Eiferman
Registration No. 51,647

Woodcock Washburn LLP
Cira Centre
2929 Arch Street, 12th Floor
Philadelphia, PA 19104-2891
Telephone: (215) 568-3100
Facsimile: (215) 568-3439